CRYPTOVERSE: A CRYPTOCURRENCY DASHBOARD (DASHBOARD)

INTRODUCTION:

The **Introduction** serves as the opening statement of the Cryptoverse project. It is the first thing users will encounter when they visit the platform, so it needs to set the stage for what the platform is and why it exists. The goal of the introduction is to quickly convey the purpose and the value that Cryptoverse brings to its users.

In this case, Cryptoverse could be a comprehensive platform designed to help users understand and interact with cryptocurrencies. It provides users with tools to track cryptocurrency prices, access blockchain technology insights, and learn about decentralized finance (DeFi). The introduction should provide a succinct and engaging explanation of these features and clearly state how the platform will make it easier for users to get involved with cryptocurrencies.

A strong introduction helps the users understand the context and encourages them to explore further. It should be welcoming, clear, and informative, addressing the primary concerns of the target audience.

Example Content:

- What is Cryptoverse?
- Why should you care about cryptocurrencies?

How can Cryptoverse help you keep up with the crypto world?

Description:

The **Description** section expands on the introduction by providing more details about the functionality, benefits, and features of Cryptoverse. This section should answer the question, "What does Cryptoverse offer, and why is it unique?"

This section will cover the core features of the platform, such as:

- **Real-time Cryptocurrency Prices:** How users can track the prices of major cryptocurrencies like Bitcoin, Ethereum, etc.
- **Educational Content:** Resources that help users learn about blockchain technology, cryptocurrency mining, smart contracts, and decentralized finance (DeFi).
- **Market Insights:** The platform could also offer additional features, such as data analysis, market trends, price volatility, and expert opinions to help users make informed decisions.
- **User Profiles or Portfolio Management:** A feature that allows users to track their investments or keep an eye on their cryptocurrency portfolio.

In this section, you should clearly highlight what users will get from the platform, the value Cryptoverse offers, and what makes it stand out from other cryptocurrency-related websites or platforms.

Example Content:

- Detailed descriptions of features like live price tracking, educational articles, and market analysis tools.
- Explanation of why these features are beneficial for both new and experienced users of cryptocurrencies.

Scenario:

The **Scenario** section is where you describe a typical user's journey on the platform. It helps users visualize how they would interact with the site, how they might benefit from it, and what kind of experience they can expect when using the platform.

For Cryptoverse, you could create scenarios for different types of users, such as:

- **Beginner Investors:** Someone who is new to the world of cryptocurrencies. They might visit Cryptoverse to get a quick understanding of cryptocurrency basics, track current prices, and explore guides on how to start investing.
- **Experienced Traders:** Users who are already familiar with cryptocurrencies and want to keep an eye on market trends, analyze price data, and stay informed on the latest blockchain news.

- **Researchers and Learners:** People interested in learning about blockchain technology and decentralized finance who may want to explore educational content and detailed reports on the future of crypto markets.

The scenario section helps make the platform feel more relatable and personal by showing how different users can derive value from using it.

Example Content:

- "John is new to cryptocurrencies and is looking for ways to invest in Bitcoin. He visits Cryptoverse, where he can see the latest price of Bitcoin and learn about blockchain technology from educational resources."

Prerequisites:

The **Prerequisites** section outlines the skills, knowledge, and tools that users may need to fully benefit from using the platform. It is important to clearly state whether there are any specific requirements to use the platform, such as familiarity with cryptocurrency terminology, basic knowledge of how cryptocurrencies work, or understanding the basics of blockchain technology.

For example:

- **Basic Crypto Knowledge:** While the platform may provide learning resources, users who understand how cryptocurrencies work will likely have a more enriching experience.
- **Web Browsing Skills:** Since Cryptoverse is a web-based platform, users need to be comfortable navigating websites.

- **Technical Prerequisites for Developers:** If the project is intended to be customizable or extendable, developers may need a basic understanding of web technologies like HTML, CSS, JavaScript, and how APIs work (in case they wish to contribute or modify the platform).

This section could also be useful if Cryptoverse is an open-source project, where developers may want to contribute. It would be important to list any tools or frameworks (like Node.js, React, etc.) that may be necessary to contribute to the development of the platform.

Example Content:

- "Basic understanding of cryptocurrencies like Bitcoin or Ethereum."
- "Familiarity with blockchain technology is helpful but not required."
- "If you're a developer, knowledge of HTML, CSS, JavaScript, and APIs will help you extend and customize the platform."

Project Structure:

The **Project Structure** section provides an overview of how the files and directories are organized in the project. If you're working on building the platform or contributing to its development, it's important to have a clear and logical file structure. It helps keep the project maintainable, scalable, and easy to navigate for future development.

For Cryptoverse, the project could be organized into several directories:

- **HTML Files:** These would contain the core content for each page (like `index.html` for the homepage, `about.html` for the About page, and `contact.html` for the Contact page).
- **CSS Files:** These files would be responsible for styling the website (e.g., `style.css` for overall page design).
- **JavaScript Files:** These would handle dynamic functionality, such as fetching live cryptocurrency prices, updating market trends, and adding interactive elements (e.g., `app.js` or `crypto.js`).

- **Assets:** This folder might include images, icons, and other resources used throughout the platform.
- **Documentation:** A README file or project documentation could explain how the project works, how to set it up, and how to contribute if it's open-source.

A well-defined structure allows developers to quickly find and modify files, making collaboration easier and development faster.

Example Content:

- `index.html` (Main homepage)
- `css/` (Contains all styling files)
- `js/` (Contains JavaScript for interactivity)
- `assets/` (Images, logos, fonts)
- `README.md` (Documentation)

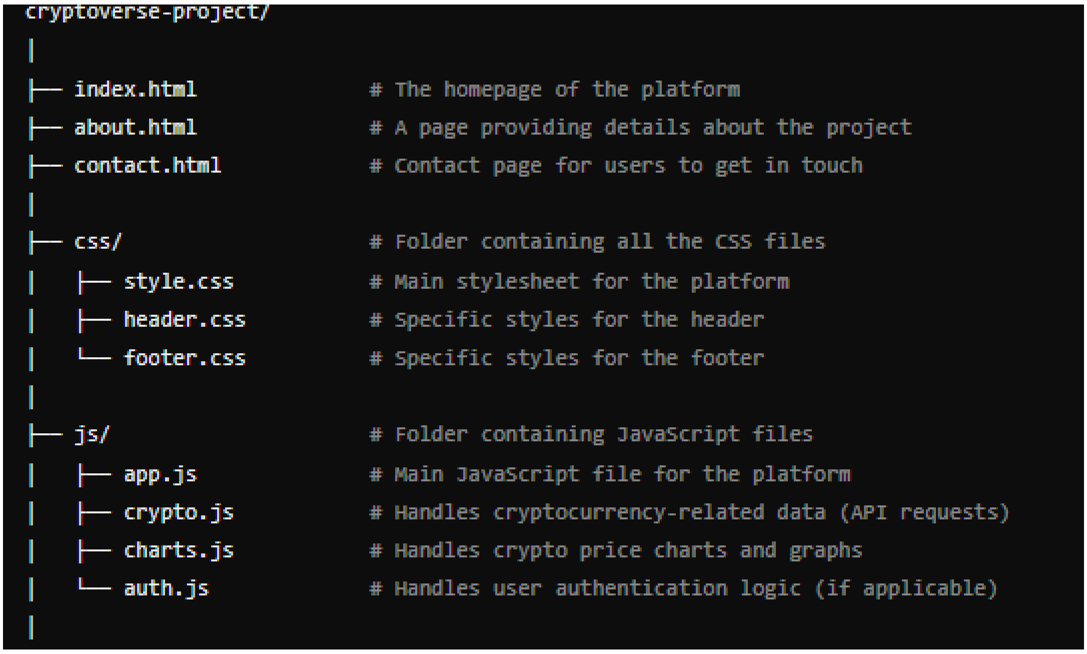
Project Flow:

The **Project Flow** outlines the user experience step-by-step, showing how users will interact with the platform. This section maps the logical flow of actions, ensuring that users can navigate the platform intuitively.

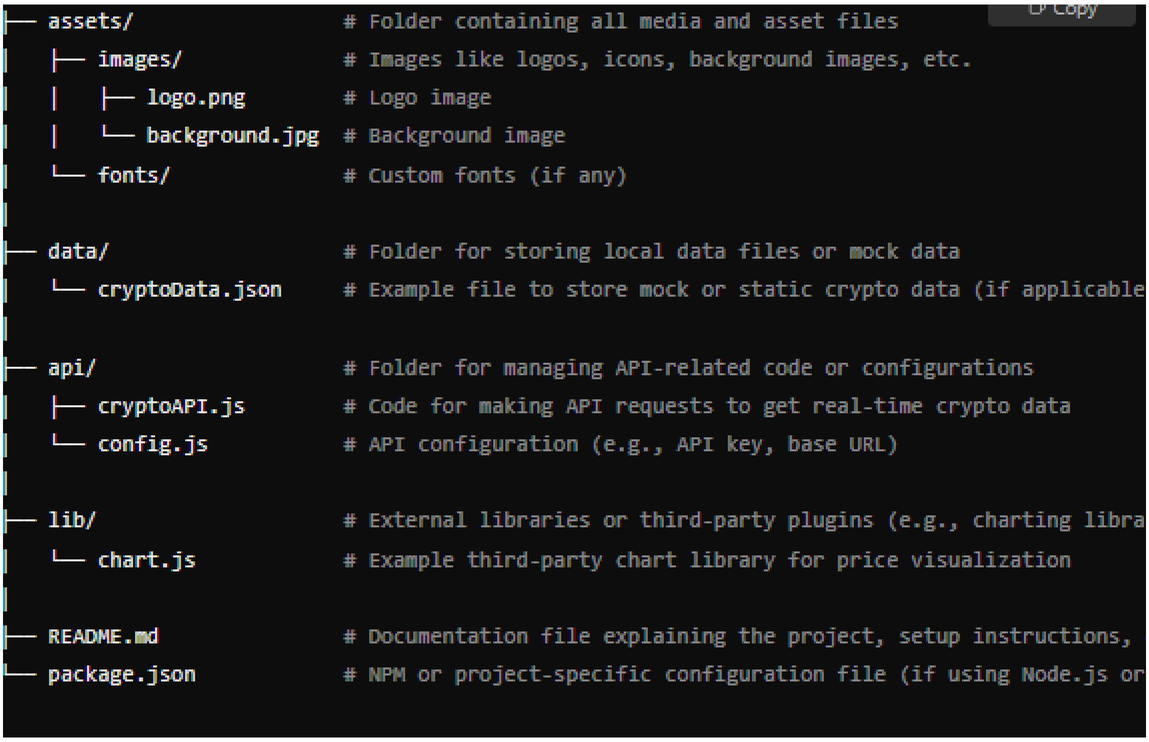
For Cryptoverse, the flow might look something like this:

1. **Homepage Visit:** The user lands on the homepage and gets a clear introduction to Cryptoverse's features, with navigation to various sections (live prices, educational content, etc.).
2. **Cryptocurrency Data:** Users can navigate to a section showing live cryptocurrency prices, market data, and trends. They might also have an option to track their favorite coins.
3. **Learning Resources:** After getting familiar with cryptocurrency data, users can move on to educational content to learn about blockchain, DeFi, smart contracts, and how to get started in crypto.
4. **Further Actions:** If users wish to track their investments, they can sign up for an account to access portfolio management features, such as creating a watchlist or getting notified when specific prices hit.
5. **Interaction with Support or Communities:** If users have questions or need assistance, they can easily find contact information or join forums/community groups.

A seamless flow ensures that users don't feel lost and can easily find what they need. It also helps in designing the user interface and improving the overall user experience.



```
cryptoverse-project/
|
├── index.html              # The homepage of the platform
├── about.html              # A page providing details about the project
├── contact.html            # Contact page for users to get in touch
|
├── css/                    # Folder containing all the CSS files
|   ├── style.css           # Main stylesheet for the platform
|   ├── header.css          # Specific styles for the header
|   └── footer.css          # Specific styles for the footer
|
├── js/                     # Folder containing JavaScript files
|   ├── app.js              # Main JavaScript file for the platform
|   ├── crypto.js           # Handles cryptocurrency-related data (API requests)
|   ├── charts.js           # Handles crypto price charts and graphs
|   └── auth.js             # Handles user authentication logic (if applicable)
|
```



```
├── assets/                 # Folder containing all media and asset files
|   ├── images/             # Images like logos, icons, background images, etc.
|   |   ├── logo.png        # Logo image
|   |   └── background.jpg  # Background image
|   └── fonts/              # Custom fonts (if any)
|
├── data/                   # Folder for storing local data files or mock data
|   └── cryptoData.json     # Example file to store mock or static crypto data (if applicable
|
├── api/                    # Folder for managing API-related code or configurations
|   ├── cryptoAPI.js        # Code for making API requests to get real-time crypto data
|   └── config.js           # API configuration (e.g., API key, base URL)
|
├── lib/                    # External libraries or third-party plugins (e.g., charting libra
|   └── chart.js            # Example third-party chart library for price visualization
|
├── README.md               # Documentation file explaining the project, setup instructions,
└── package.json            # NPM or project-specific configuration file (if using Node.js or
```

Creating Cryptocurrency Component:

The **Cryptocurrency Component** is a core feature of Cryptoverse, enabling the platform to provide live data about various cryptocurrencies. This component will be responsible for fetching data from external sources (usually APIs) and displaying it in real-time.

- The component will likely interact with a cryptocurrency data provider, such as CoinGecko, CoinMarketCap, or a custom API.
- The data will include prices, market cap, trading volume, price fluctuations, etc., and will be presented in an easy-to-understand format like charts, tables, or cards.
- Users will be able to view up-to-date information for popular cryptocurrencies such as Bitcoin (BTC), Ethereum (ETH), Litecoin (LTC), and others.

The component is highly dynamic and will need to update frequently, potentially every few seconds, to ensure the data is current. It may also feature additional functionality, such as sorting, filtering, and providing price alerts or market analysis.

Example Content:

- A card showing the price, market cap, and 24-hour price change for Bitcoin.
- A chart displaying the historical price trend of Ethereum.
- Interactive buttons to view price data for different cryptocurrencies.

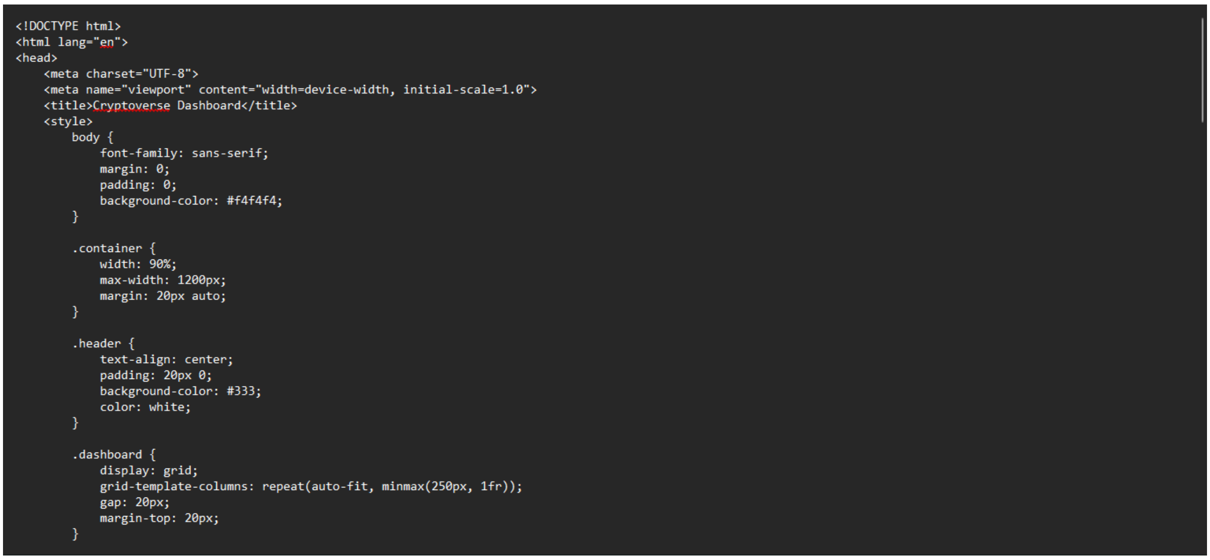
Creating Homepage:

The **Homepage** is the first thing users will see when they visit Cryptoverse. It should be welcoming, easy to navigate, and provide an overview of what Cryptoverse offers. The homepage is the user's entry point to the platform, so it needs to provide access to all major features and guide users through their next steps.

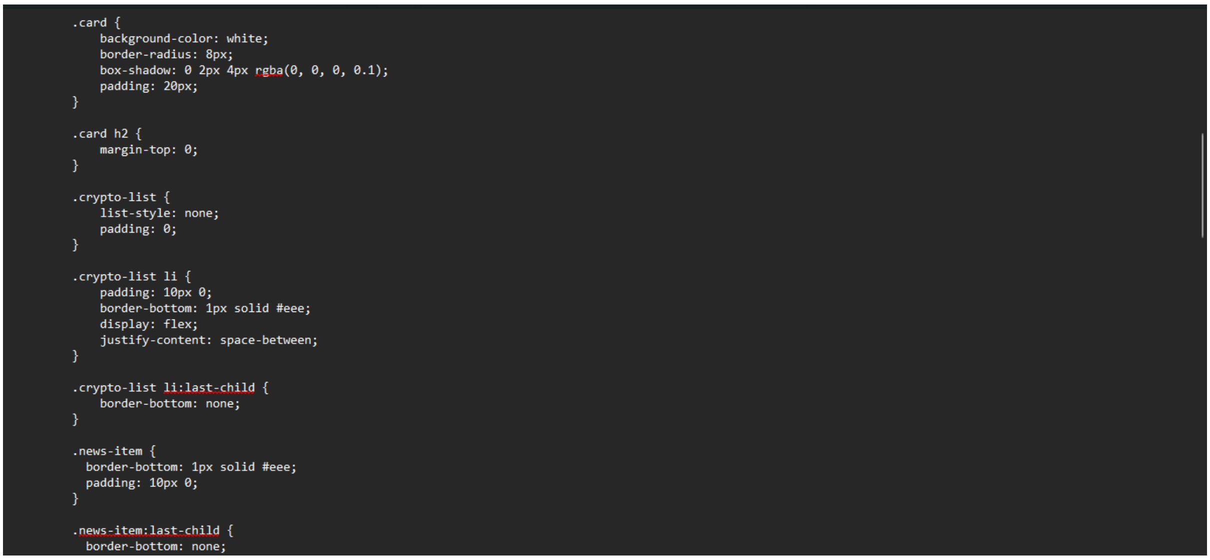
- **Navigation:** The homepage should have a clear, easy-to-use navigation bar that lets users access all the main sections of the site, such as cryptocurrency price tracking, educational content, and market analysis.
- **Introduction:** The homepage should briefly introduce Cryptoverse and explain its purpose. This could include a tagline, a call-to-action (CTA), and a quick summary of the platform's offerings.

- **Visual Appeal:** The homepage should be visually engaging, with a clean, modern design that reflects the theme of cryptocurrencies and blockchain.
- **Featured Sections:** Highlight key features of the platform, like live price tracking, crypto news, and learning resources, in an easy-to-digest format.

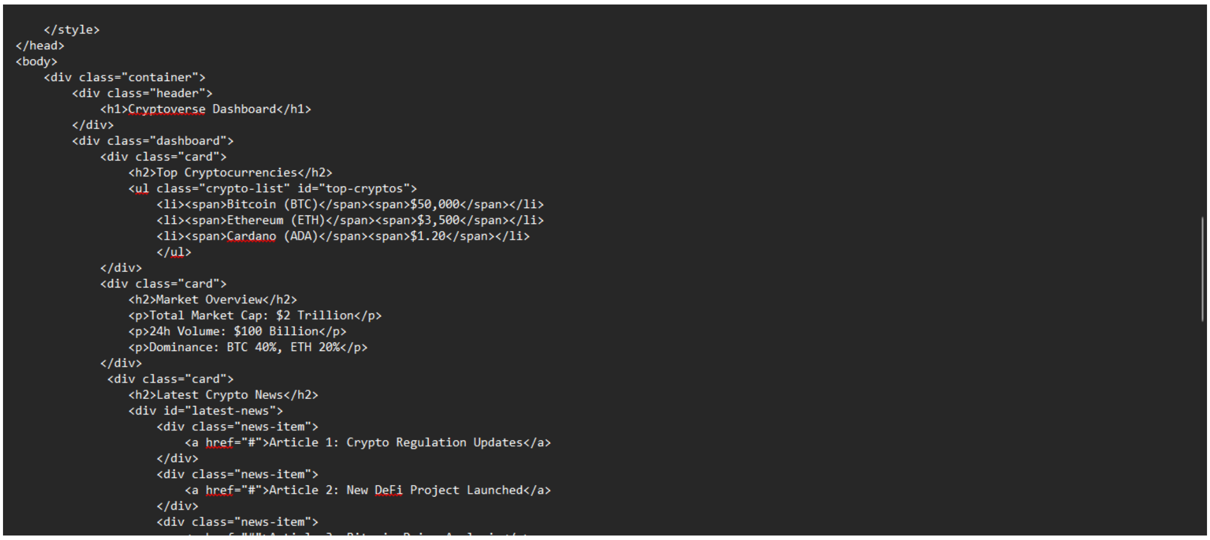
Coding:



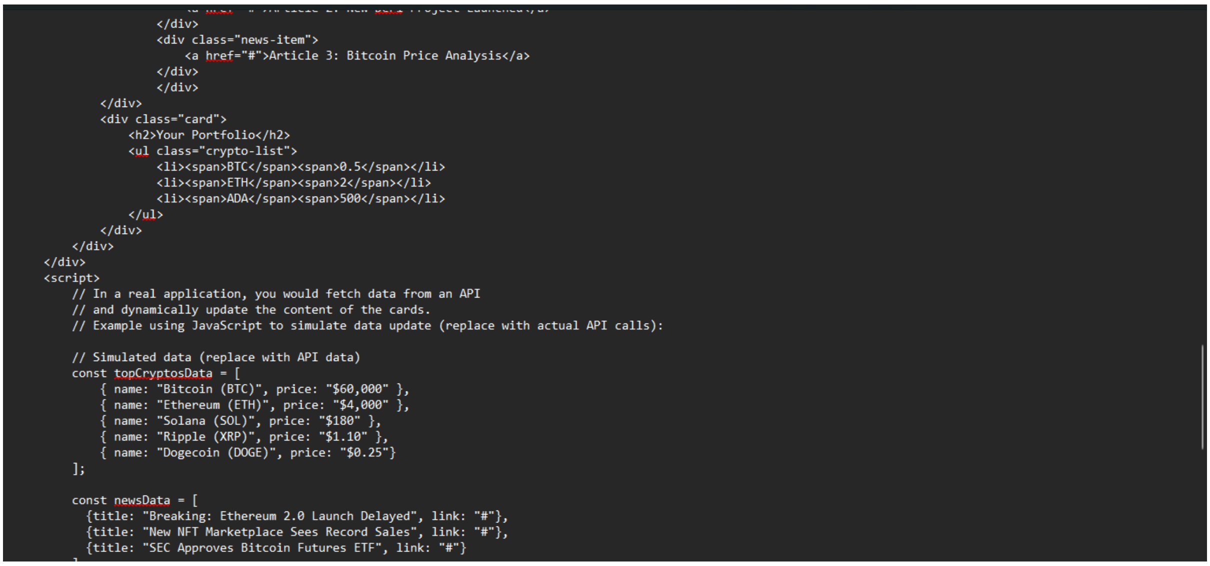
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Cryptoverse Dashboard</title>
    <style>
        body {
            font-family: sans-serif;
            margin: 0;
            padding: 0;
            background-color: #f4f4f4;
        }

        .container {
            width: 90%;
            max-width: 1200px;
            margin: 20px auto;
        }

        .header {
            text-align: center;
            padding: 20px 0;
            background-color: #333;
            color: white;
        }

        .dashboard {
            display: grid;
            grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));
            gap: 20px;
            margin-top: 20px;
        }
```



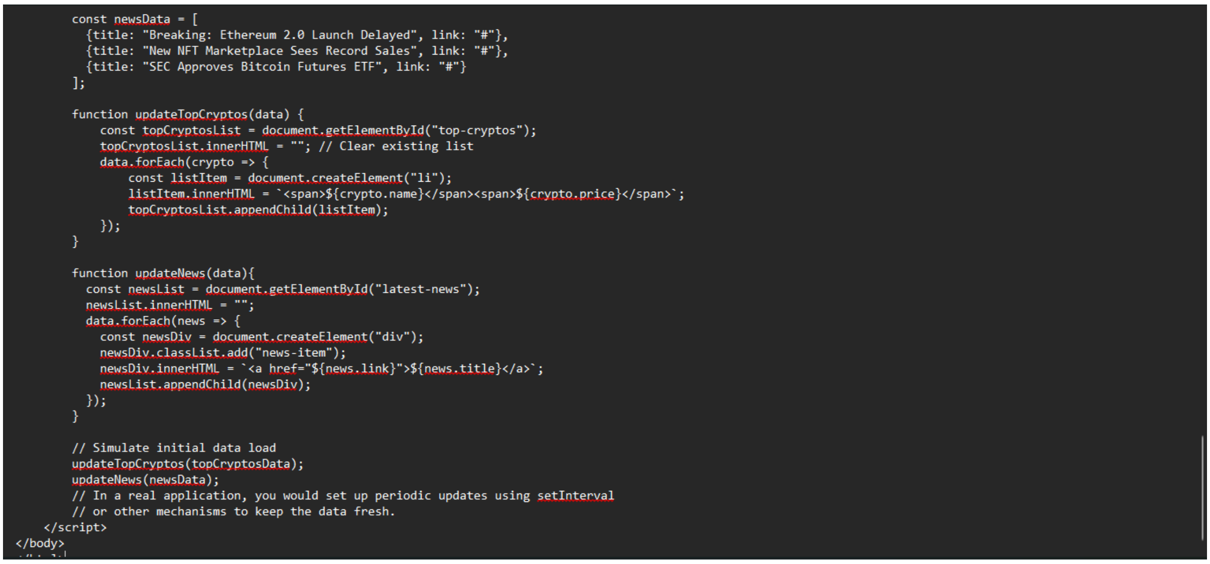
```
        .card {
            background-color: white;
            border-radius: 8px;
            box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);
            padding: 20px;
        }

        .card h2 {
            margin-top: 0;
        }

        .crypto-list {
            list-style: none;
            padding: 0;
        }

        .crypto-list li {
            padding: 10px 0;
            border-bottom: 1px solid #eee;
            display: flex;
            justify-content: space-between;
        }

        .crypto-list li:last-child {
            border-bottom: none;
        }

        .news-item {
          border-bottom: 1px solid #eee;
          padding: 10px 0;
        }

        .news-item:last-child {
          border-bottom: none;
```



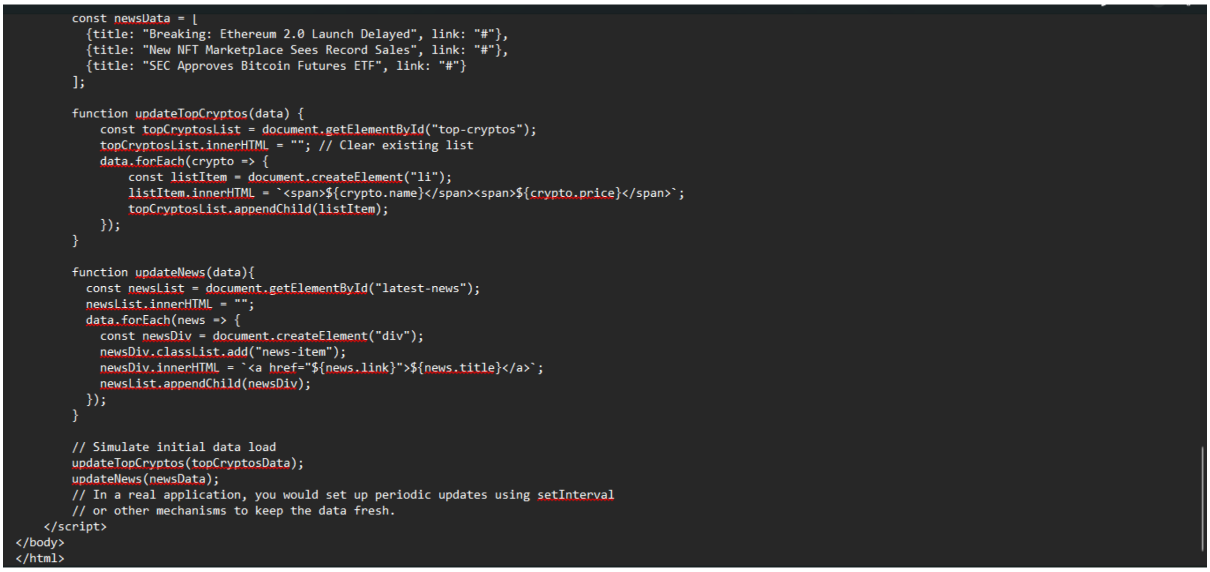
```
    </style>
</head>
<body>
    <div class="container">
        <div class="header">
            <h1>Cryptoverse Dashboard</h1>
        </div>
        <div class="dashboard">
            <div class="card">
                <h2>Top Cryptocurrencies</h2>
                <ul class="crypto-list" id="top-cryptos">
                    <li><span>Bitcoin (BTC)</span><span>$50,000</span></li>
                    <li><span>Ethereum (ETH)</span><span>$3,500</span></li>
                    <li><span>Cardano (ADA)</span><span>$1.20</span></li>
                    </ul>
            </div>
            <div class="card">
                <h2>Market Overview</h2>
                <p>Total Market Cap: $2 Trillion</p>
                <p>24h Volume: $100 Billion</p>
                <p>Dominance: BTC 40%, ETH 20%</p>
            </div>
             <div class="card">
                <h2>Latest Crypto News</h2>
                <div id="latest-news">
                    <div class="news-item">
                        <a href="#">Article 1: Crypto Regulation Updates</a>
                    </div>
                    <div class="news-item">
                        <a href="#">Article 2: New DeFi Project Launched</a>
                    </div>
                    <div class="news-item">
```



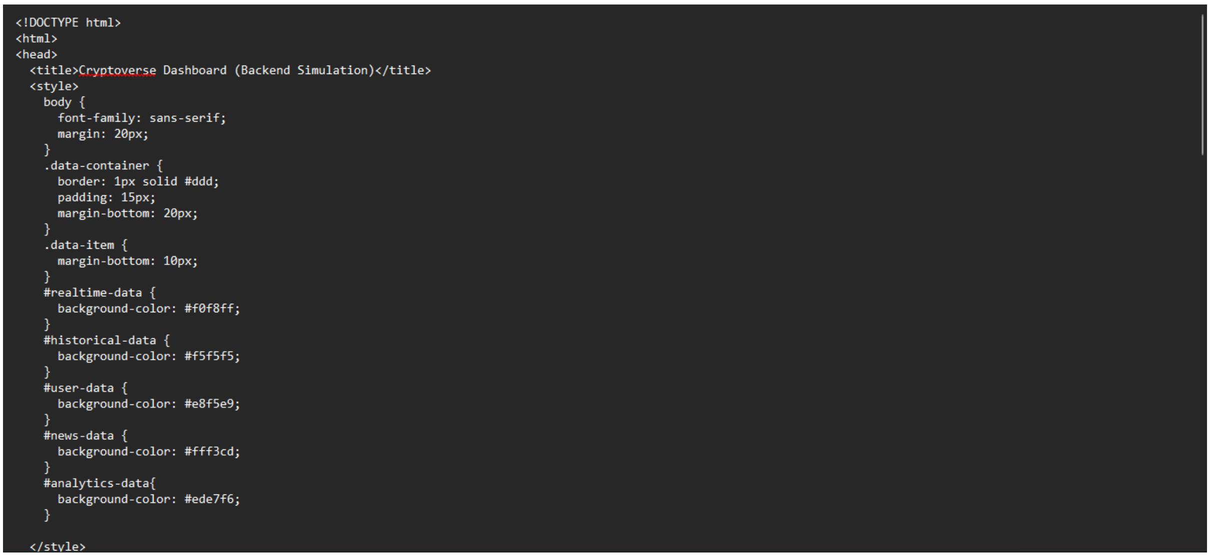
```
                    </div>
                    <div class="news-item">
                        <a href="#">Article 3: Bitcoin Price Analysis</a>
                    </div>
                    </div>
            </div>
            <div class="card">
                <h2>Your Portfolio</h2>
                <ul class="crypto-list">
                    <li><span>BTC</span><span>0.5</span></li>
                    <li><span>ETH</span><span>2</span></li>
                    <li><span>ADA</span><span>500</span></li>
                </ul>
            </div>
        </div>
    </div>
    <script>
        // In a real application, you would fetch data from an API
        // and dynamically update the content of the cards.
        // Example using JavaScript to simulate data update (replace with actual API calls):

        // Simulated data (replace with API data)
        const topCryptosData = [
            { name: "Bitcoin (BTC)", price: "$60,000" },
            { name: "Ethereum (ETH)", price: "$4,000" },
            { name: "Solana (SOL)", price: "$180" },
            { name: "Ripple (XRP)", price: "$1.10" },
            { name: "Dogecoin (DOGE)", price: "$0.25"}
        ];

        const newsData = [
          {title: "Breaking: Ethereum 2.0 Launch Delayed", link: "#"},
          {title: "New NFT Marketplace Sees Record Sales", link: "#"},
          {title: "SEC Approves Bitcoin Futures ETF", link: "#"}
```



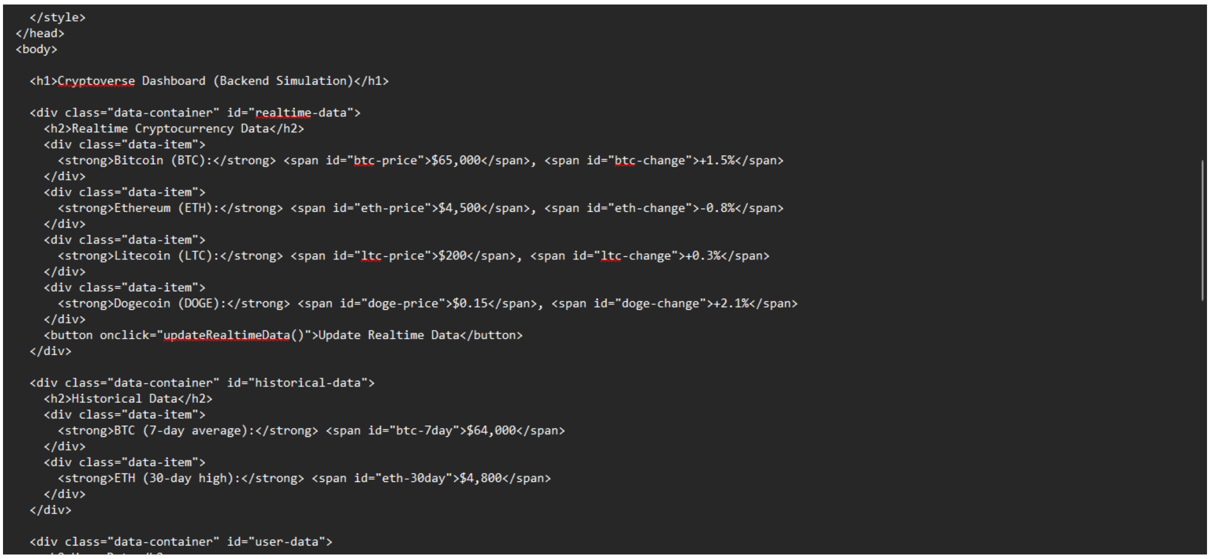
```
        const newsData = [
          {title: "Breaking: Ethereum 2.0 Launch Delayed", link: "#"},
          {title: "New NFT Marketplace Sees Record Sales", link: "#"},
          {title: "SEC Approves Bitcoin Futures ETF", link: "#"}
        ];

        function updateTopCryptos(data) {
            const topCryptosList = document.getElementById("top-cryptos");
            topCryptosList.innerHTML = ""; // Clear existing list
            data.forEach(crypto => {
                const listItem = document.createElement("li");
                listItem.innerHTML = `<span>${crypto.name}</span><span>${crypto.price}</span>`;
                topCryptosList.appendChild(listItem);
            });
        }

        function updateNews(data){
          const newsList = document.getElementById("latest-news");
          newsList.innerHTML = "";
          data.forEach(news => {
            const newsDiv = document.createElement("div");
            newsDiv.classList.add("news-item");
            newsDiv.innerHTML = `<a href="${news.link}">${news.title}</a>`;
            newsList.appendChild(newsDiv);
          });
        }

        // Simulate initial data load
        updateTopCryptos(topCryptosData);
        updateNews(newsData);
        // In a real application, you would set up periodic updates using setInterval
        // or other mechanisms to keep the data fresh.
    </script>
</body>
```



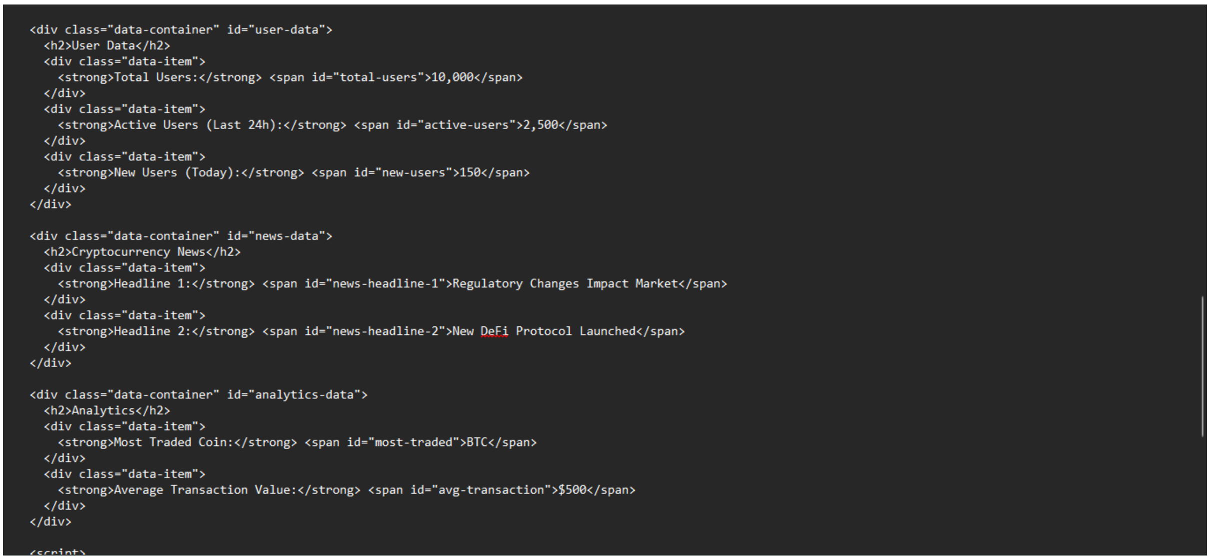
```
        const newsData = [
          {title: "Breaking: Ethereum 2.0 Launch Delayed", link: "#"},
          {title: "New NFT Marketplace Sees Record Sales", link: "#"},
          {title: "SEC Approves Bitcoin Futures ETF", link: "#"}
        ];

        function updateTopCryptos(data) {
            const topCryptosList = document.getElementById("top-cryptos");
            topCryptosList.innerHTML = ""; // Clear existing list
            data.forEach(crypto => {
                const listItem = document.createElement("li");
                listItem.innerHTML = `<span>${crypto.name}</span><span>${crypto.price}</span>`;
                topCryptosList.appendChild(listItem);
            });
        }

        function updateNews(data){
          const newsList = document.getElementById("latest-news");
          newsList.innerHTML = "";
          data.forEach(news => {
            const newsDiv = document.createElement("div");
            newsDiv.classList.add("news-item");
            newsDiv.innerHTML = `<a href="${news.link}">${news.title}</a>`;
            newsList.appendChild(newsDiv);
          });
        }

        // Simulate initial data load
        updateTopCryptos(topCryptosData);
        updateNews(newsData);
        // In a real application, you would set up periodic updates using setInterval
        // or other mechanisms to keep the data fresh.
    </script>
</body>
</html>
```



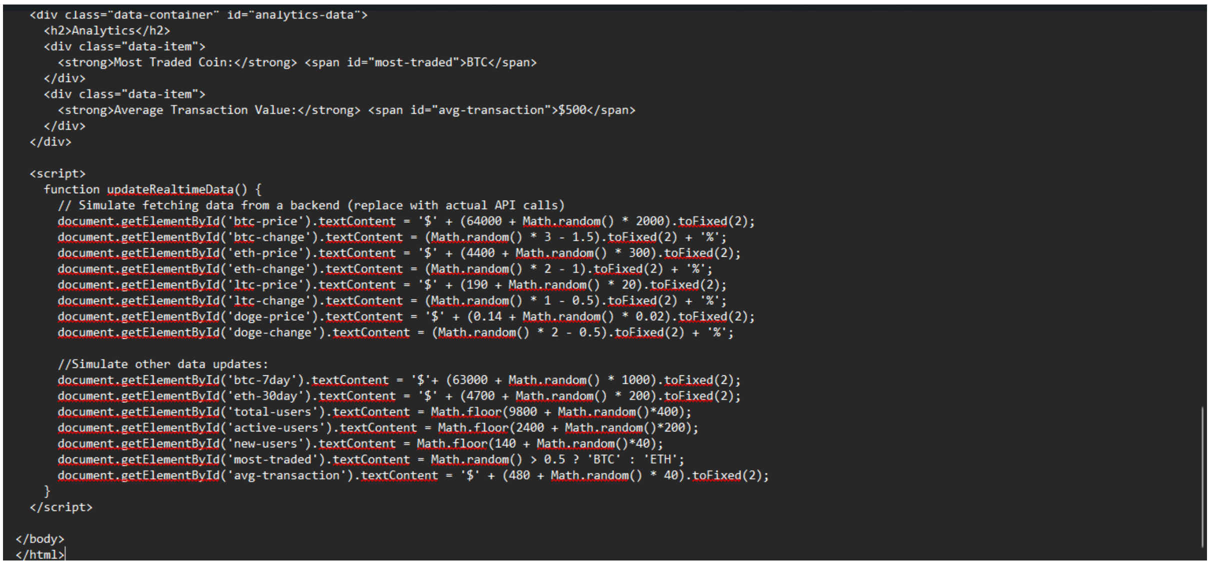
```
<!DOCTYPE html>
<html>
<head>
  <title>Cryptoverse Dashboard (Backend Simulation)</title>
  <style>
    body {
      font-family: sans-serif;
      margin: 20px;
    }
    .data-container {
      border: 1px solid #ddd;
      padding: 15px;
      margin-bottom: 20px;
    }
    .data-item {
      margin-bottom: 10px;
    }
    #realtime-data {
      background-color: #f0f8ff;
    }
    #historical-data {
      background-color: #f5f5f5;
    }
    #user-data {
      background-color: #e8f5e9;
    }
    #news-data {
      background-color: #fff3cd;
    }
    #analytics-data{
      background-color: #ede7f6;
    }

  </style>
```



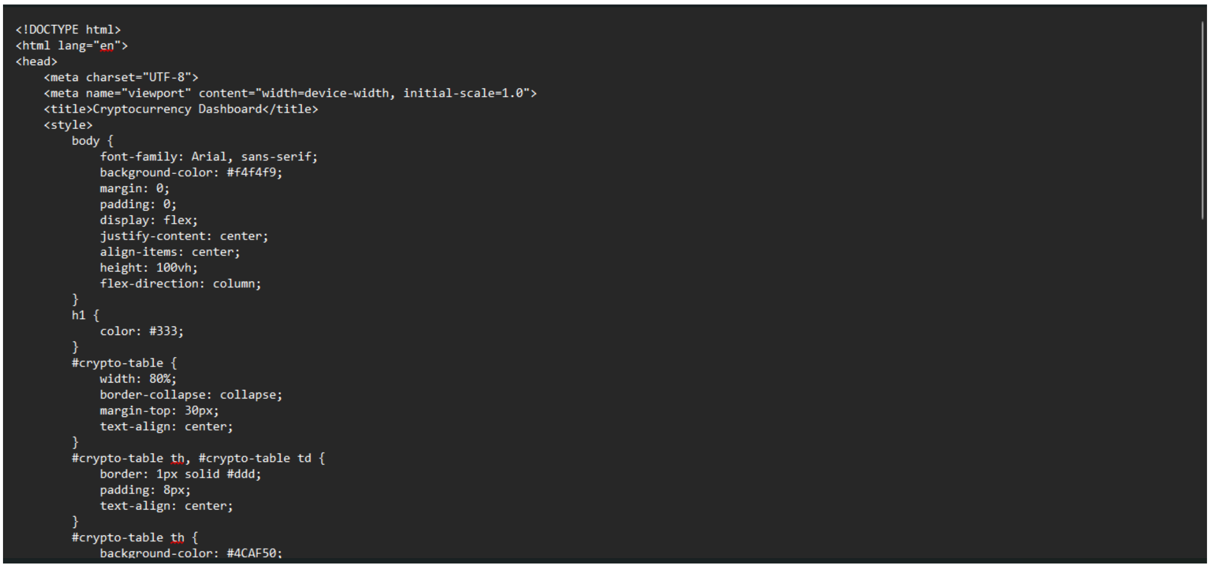
```
  </style>
</head>
<body>

  <h1>Cryptoverse Dashboard (Backend Simulation)</h1>

  <div class="data-container" id="realtime-data">
    <h2>Realtime Cryptocurrency Data</h2>
    <div class="data-item">
      <strong>Bitcoin (BTC):</strong> <span id="btc-price">$65,000</span>, <span id="btc-change">+1.5%</span>
    </div>
    <div class="data-item">
      <strong>Ethereum (ETH):</strong> <span id="eth-price">$4,500</span>, <span id="eth-change">-0.8%</span>
    </div>
    <div class="data-item">
      <strong>Litecoin (LTC):</strong> <span id="ltc-price">$200</span>, <span id="ltc-change">+0.3%</span>
    </div>
    <div class="data-item">
      <strong>Dogecoin (DOGE):</strong> <span id="doge-price">$0.15</span>, <span id="doge-change">+2.1%</span>
    </div>
    <button onclick="updateRealtimeData()">Update Realtime Data</button>
  </div>

  <div class="data-container" id="historical-data">
    <h2>Historical Data</h2>
    <div class="data-item">
      <strong>BTC (7-day average):</strong> <span id="btc-7day">$64,000</span>
    </div>
    <div class="data-item">
      <strong>ETH (30-day high):</strong> <span id="eth-30day">$4,800</span>
    </div>
  </div>

  <div class="data-container" id="user-data">
```



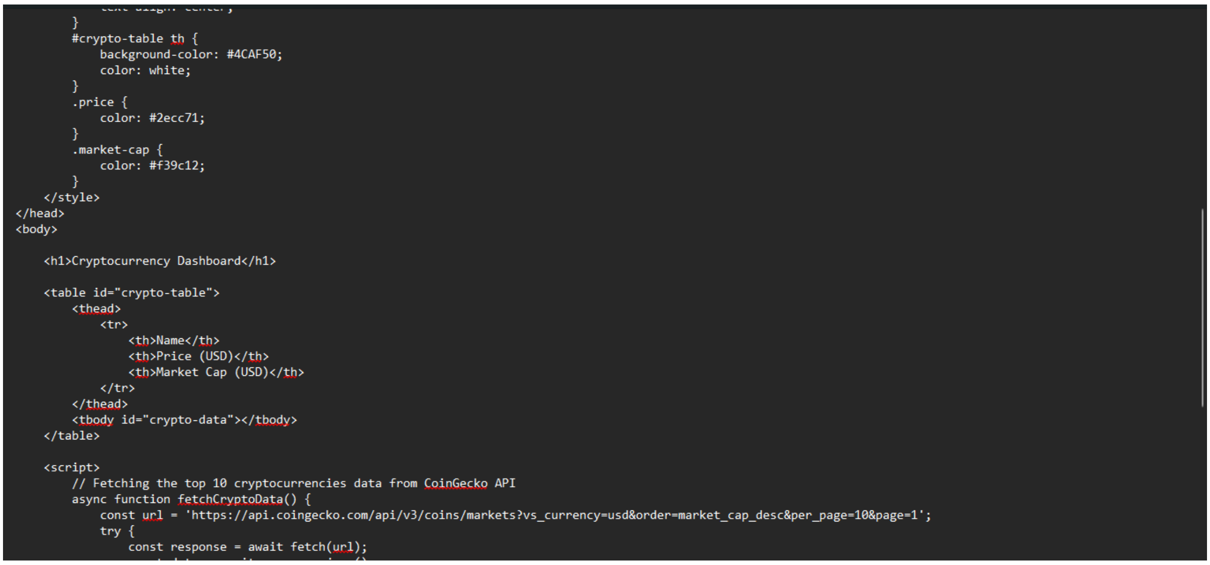
```
  <div class="data-container" id="user-data">
    <h2>User Data</h2>
    <div class="data-item">
      <strong>Total Users:</strong> <span id="total-users">10,000</span>
    </div>
    <div class="data-item">
      <strong>Active Users (Last 24h):</strong> <span id="active-users">2,500</span>
    </div>
    <div class="data-item">
      <strong>New Users (Today):</strong> <span id="new-users">150</span>
    </div>
  </div>

  <div class="data-container" id="news-data">
    <h2>Cryptocurrency News</h2>
    <div class="data-item">
      <strong>Headline 1:</strong> <span id="news-headline-1">Regulatory Changes Impact Market</span>
    </div>
    <div class="data-item">
      <strong>Headline 2:</strong> <span id="news-headline-2">New DeFi Protocol Launched</span>
    </div>
  </div>

  <div class="data-container" id="analytics-data">
    <h2>Analytics</h2>
    <div class="data-item">
      <strong>Most Traded Coin:</strong> <span id="most-traded">BTC</span>
    </div>
    <div class="data-item">
      <strong>Average Transaction Value:</strong> <span id="avg-transaction">$500</span>
    </div>
  </div>

  <script>
```



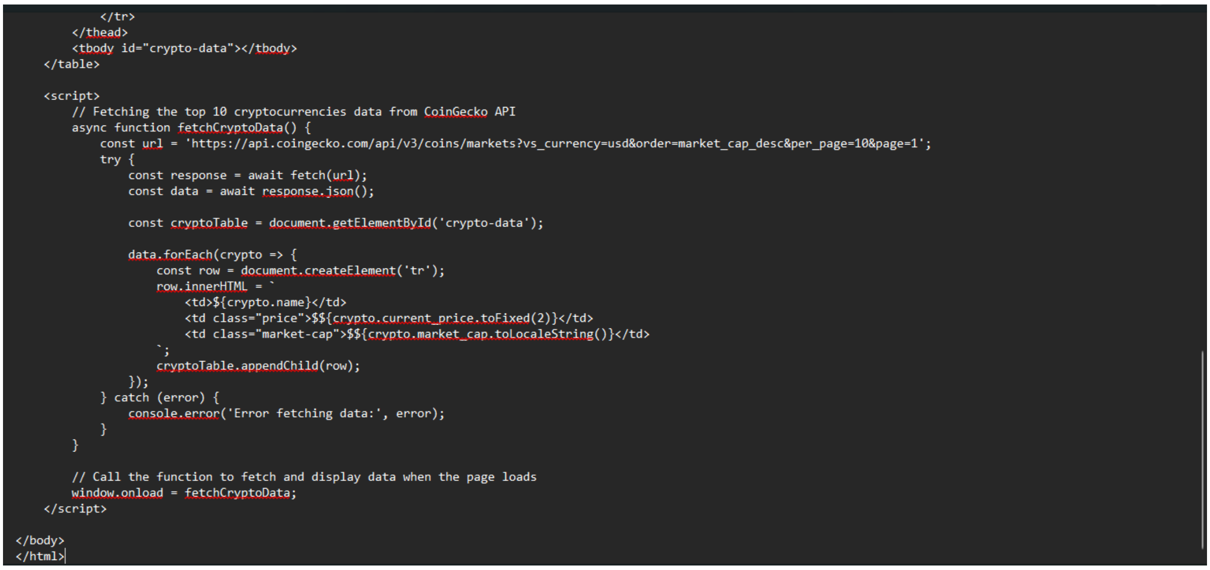
```
  <div class="data-container" id="analytics-data">
    <h2>Analytics</h2>
    <div class="data-item">
      <strong>Most Traded Coin:</strong> <span id="most-traded">BTC</span>
    </div>
    <div class="data-item">
      <strong>Average Transaction Value:</strong> <span id="avg-transaction">$500</span>
    </div>
  </div>

  <script>
    function updateRealtimeData() {
      // Simulate fetching data from a backend (replace with actual API calls)
      document.getElementById('btc-price').textContent = '$' + (64000 + Math.random() * 2000).toFixed(2);
      document.getElementById('btc-change').textContent = (Math.random() * 3 - 1.5).toFixed(2) + '%';
      document.getElementById('eth-price').textContent = '$' + (4400 + Math.random() * 300).toFixed(2);
      document.getElementById('eth-change').textContent = (Math.random() * 2 - 1).toFixed(2) + '%';
      document.getElementById('ltc-price').textContent = '$' + (190 + Math.random() * 20).toFixed(2);
      document.getElementById('ltc-change').textContent = (Math.random() * 1 - 0.5).toFixed(2) + '%';
      document.getElementById('doge-price').textContent = '$' + (0.14 + Math.random() * 0.02).toFixed(2);
      document.getElementById('doge-change').textContent = (Math.random() * 2 - 0.5).toFixed(2) + '%';

      //Simulate other data updates:
      document.getElementById('btc-7day').textContent = '$'+ (63000 + Math.random() * 1000).toFixed(2);
      document.getElementById('eth-30day').textContent = '$' + (4700 + Math.random() * 200).toFixed(2);
      document.getElementById('total-users').textContent = Math.floor(9800 + Math.random()*400);
      document.getElementById('active-users').textContent = Math.floor(2400 + Math.random()*200);
      document.getElementById('new-users').textContent = Math.floor(140 + Math.random()*40);
      document.getElementById('most-traded').textContent = Math.random() > 0.5 ? 'BTC' : 'ETH';
      document.getElementById('avg-transaction').textContent = '$' + (480 + Math.random() * 40).toFixed(2);
    }
  </script>

</body>
</html>
```



```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Cryptocurrency Dashboard</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #f4f4f9;
            margin: 0;
            padding: 0;
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
            flex-direction: column;
        }
        h1 {
            color: #333;
        }
        #crypto-table {
            width: 80%;
            border-collapse: collapse;
            margin-top: 30px;
            text-align: center;
        }
        #crypto-table th, #crypto-table td {
            border: 1px solid #ddd;
            padding: 8px;
            text-align: center;
        }
        #crypto-table th {
            background-color: #4CAF50;
```

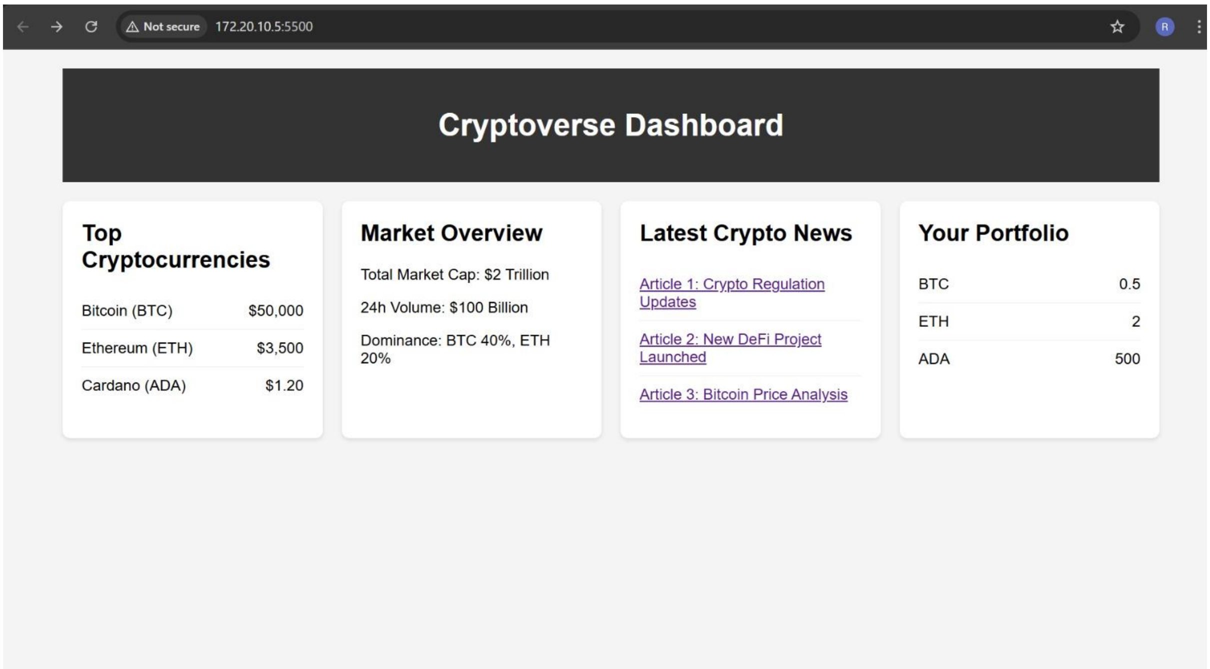


```
            text-align: center;
        }
        #crypto-table th {
            background-color: #4CAF50;
            color: white;
        }
        .price {
            color: #2ecc71;
        }
        .market-cap {
            color: #f39c12;
        }
    </style>
</head>
<body>

    <h1>Cryptocurrency Dashboard</h1>

    <table id="crypto-table">
        <thead>
            <tr>
                <th>Name</th>
                <th>Price (USD)</th>
                <th>Market Cap (USD)</th>
            </tr>
        </thead>
        <tbody id="crypto-data"></tbody>
    </table>

    <script>
        // Fetching the top 10 cryptocurrencies data from CoinGecko API
        async function fetchCryptoData() {
            const url = 'https://api.coingecko.com/api/v3/coins/markets?vs_currency=usd&order=market_cap_desc&per_page=10&page=1';
            try {
                const response = await fetch(url);
```



```
            </tr>
        </thead>
        <tbody id="crypto-data"></tbody>
    </table>

    <script>
        // Fetching the top 10 cryptocurrencies data from CoinGecko API
        async function fetchCryptoData() {
            const url = 'https://api.coingecko.com/api/v3/coins/markets?vs_currency=usd&order=market_cap_desc&per_page=10&page=1';
            try {
                const response = await fetch(url);
                const data = await response.json();

                const cryptoTable = document.getElementById('crypto-data');

                data.forEach(crypto => {
                    const row = document.createElement('tr');
                    row.innerHTML = `
                        <td>${crypto.name}</td>
                        <td class="price">$${crypto.current_price.toFixed(2)}</td>
                        <td class="market-cap">$${crypto.market_cap.toLocaleString()}</td>
                    `;
                    cryptoTable.appendChild(row);
                });
            } catch (error) {
                console.error('Error fetching data:', error);
            }
        }

        // Call the function to fetch and display data when the page loads
        window.onload = fetchCryptoData;
    </script>

</body>
</html>
```

Output:



Cryptoverse Dashboard

Top Cryptocurrencies

Bitcoin (BTC)	$50,000
Ethereum (ETH)	$3,500
Cardano (ADA)	$1.20

Market Overview

Total Market Cap: $2 Trillion

24h Volume: $100 Billion

Dominance: BTC 40%, ETH 20%

Latest Crypto News

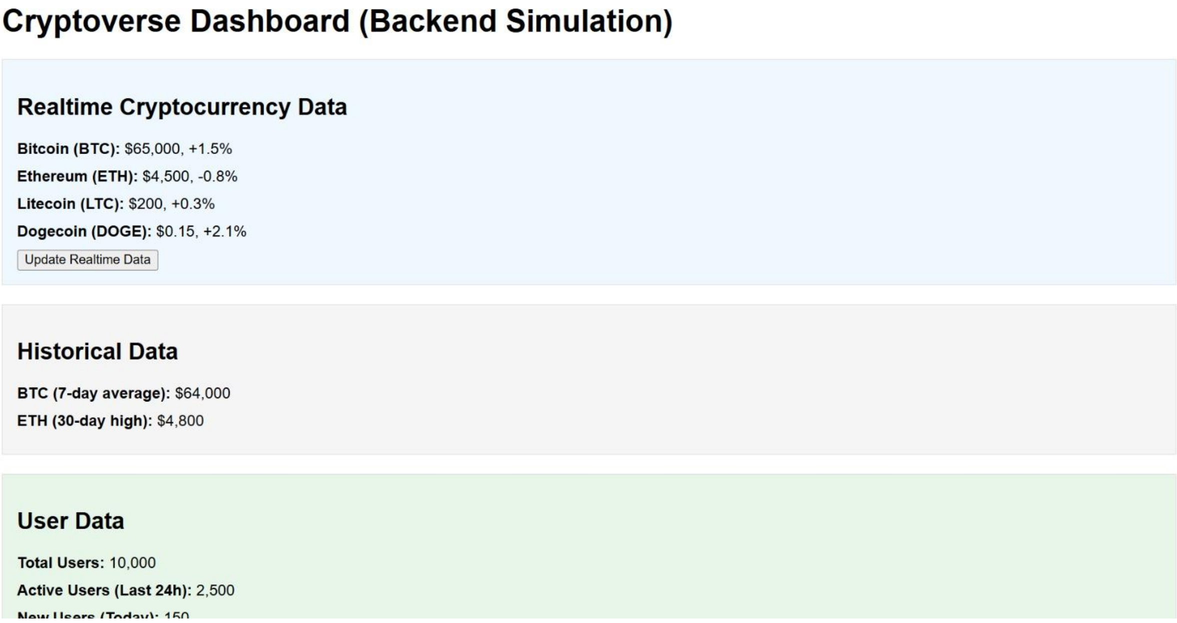
Article 1: Crypto Regulation Updates

Article 2: New DeFi Project Launched

Article 3: Bitcoin Price Analysis

Your Portfolio

BTC	0.5
ETH	2
ADA	500



Cryptoverse Dashboard (Backend Simulation)

Realtime Cryptocurrency Data

Bitcoin (BTC): $65,000, +1.5%

Ethereum (ETH): $4,500, -0.8%

Litecoin (LTC): $200, +0.3%

Dogecoin (DOGE): $0.15, +2.1%

Historical Data

BTC (7-day average): $64,000

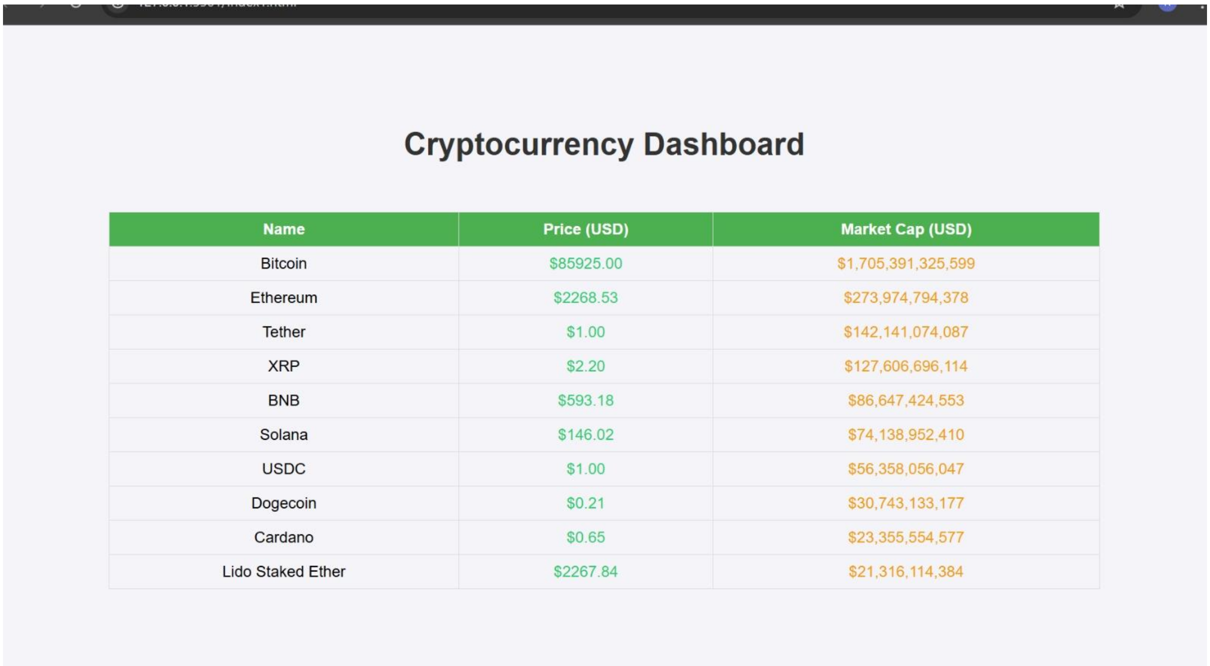
ETH (30-day high): $4,800

User Data

Total Users: 10,000

Active Users (Last 24h): 2,500

New Users (Today): 150

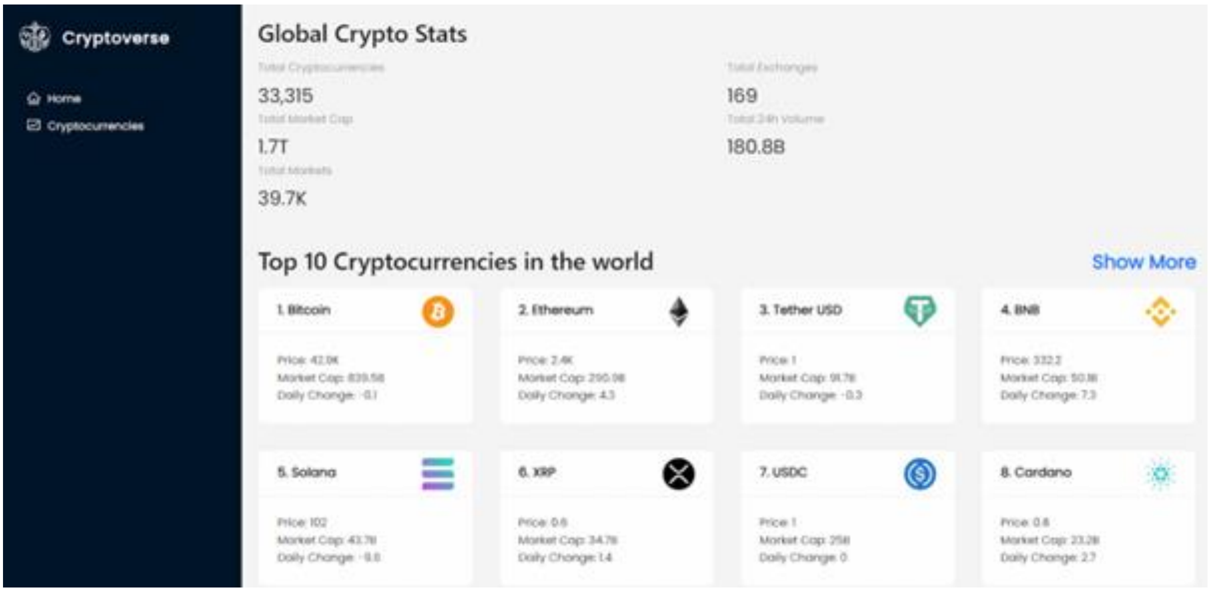


Cryptocurrency Dashboard

Name	Price (USD)	Market Cap (USD)
Bitcoin	$85925.00	$1,705,391,325,599
Ethereum	$2268.53	$273,974,794,378
Tether	$1.00	$142,141,074,087
XRP	$2.20	$127,606,696,114
BNB	$593.18	$86,647,424,553
Solana	$146.02	$74,138,952,410
USDC	$1.00	$56,358,056,047
Dogecoin	$0.21	$30,743,133,177
Cardano	$0.65	$23,355,554,577
Lido Staked Ether	$2267.84	$21,316,114,384

User Interface snips:

Homepage: This pages consists of stats of global crypto like total cryptocurrencies, total exchanges, market cap etc. Also consist of top 10 cryptocurrencies in the world.



Global Crypto Stats

Total Cryptocurrencies
33,315

Total Exchanges
169

Total Market Cap
1.7T

Total 24h Volume
180.8B

Total Markets
39.7K

Top 10 Cryptocurrencies in the world

1. Bitcoin
Price: 42.0K
Market Cap: 839.5B
Daily Change: -0.1

2. Ethereum
Price: 2.4K
Market Cap: 290.0B
Daily Change: 4.3

3. Tether USD
Price: 1
Market Cap: 91.7B
Daily Change: -0.3

4. BNB
Price: 332.2
Market Cap: 50.1B
Daily Change: 7.3

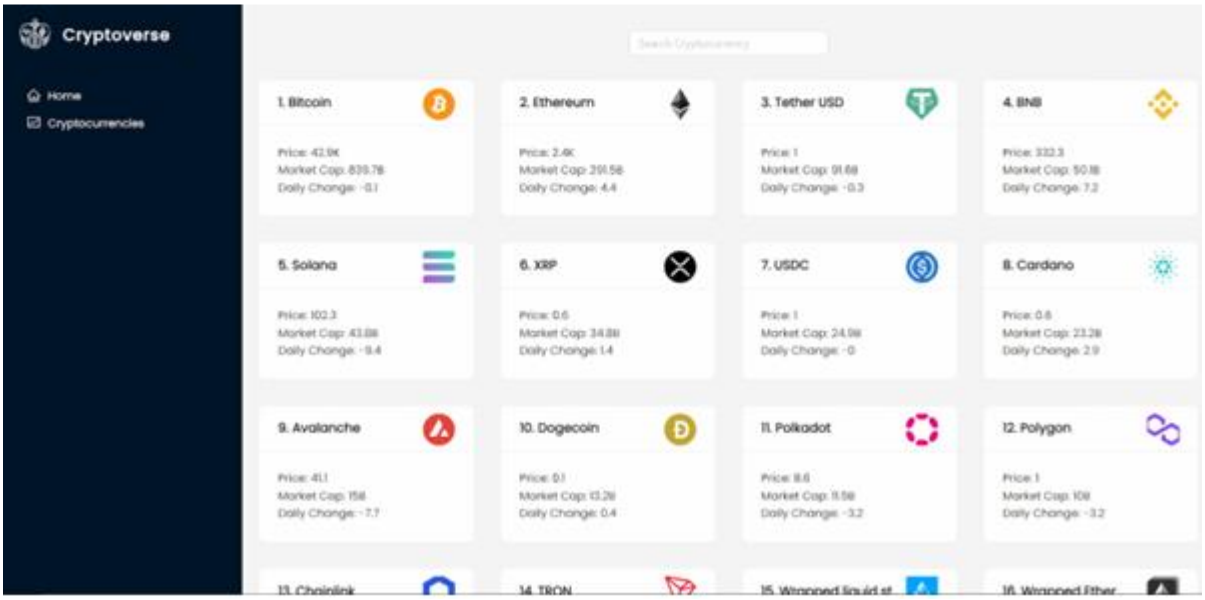
5. Solana
Price: 102
Market Cap: 43.7B
Daily Change: -9.8

6. XRP
Price: 0.6
Market Cap: 34.7B
Daily Change: 1.4

7. USDC
Price: 1
Market Cap: 25B
Daily Change: 0

8. Cardano
Price: 0.6
Market Cap: 23.2B
Daily Change: 2.7

Crypto currencies page : This pages contains all cryptocurrencies which are currently in flow in the world. There is also a search feature where users can search and find out about their desired cryptocurrency.



Cryptoverse

Home

Cryptocurrencies

1. Bitcoin

Price: 42.9K
Market Cap: 839.7B
Daily Change: -0.1

2. Ethereum

Price: 2.4K
Market Cap: 291.5B
Daily Change: 4.4

3. Tether USD

Price: 1
Market Cap: 91.6B
Daily Change: -0.3

4. BNB

Price: 332.3
Market Cap: 50.1B
Daily Change: 7.2

5. Solana

Price: 102.3
Market Cap: 43.8B
Daily Change: -9.4

6. XRP

Price: 0.6
Market Cap: 34.8B
Daily Change: 1.4

7. USDC

Price: 1
Market Cap: 24.9B
Daily Change: -0

8. Cardano

Price: 0.6
Market Cap: 23.2B
Daily Change: 2.9

9. Avalanche

Price: 41.1
Market Cap: 15B
Daily Change: -7.7

10. Dogecoin

Price: 0.1
Market Cap: 13.2B
Daily Change: 0.4

11. Polkadot

Price: 8.6
Market Cap: 11.5B
Daily Change: -3.2

12. Polygon

Price: 1
Market Cap: 10B
Daily Change: -3.2

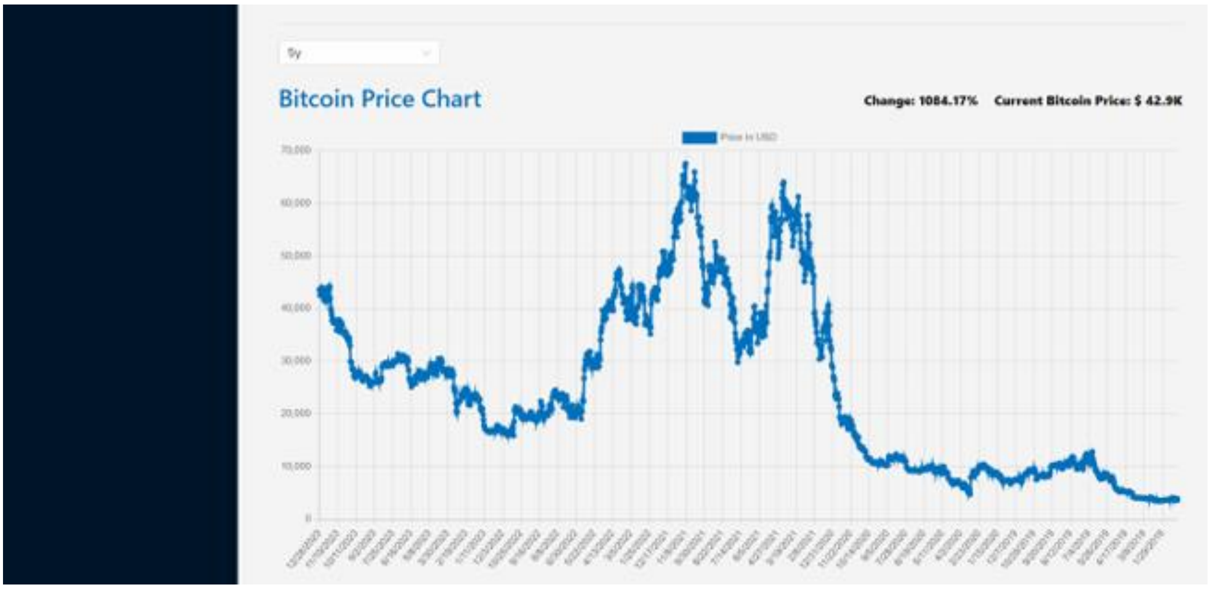
13. Chainlink

14. TRON

15. Wrapped liquid st

16. Wrapped Ether

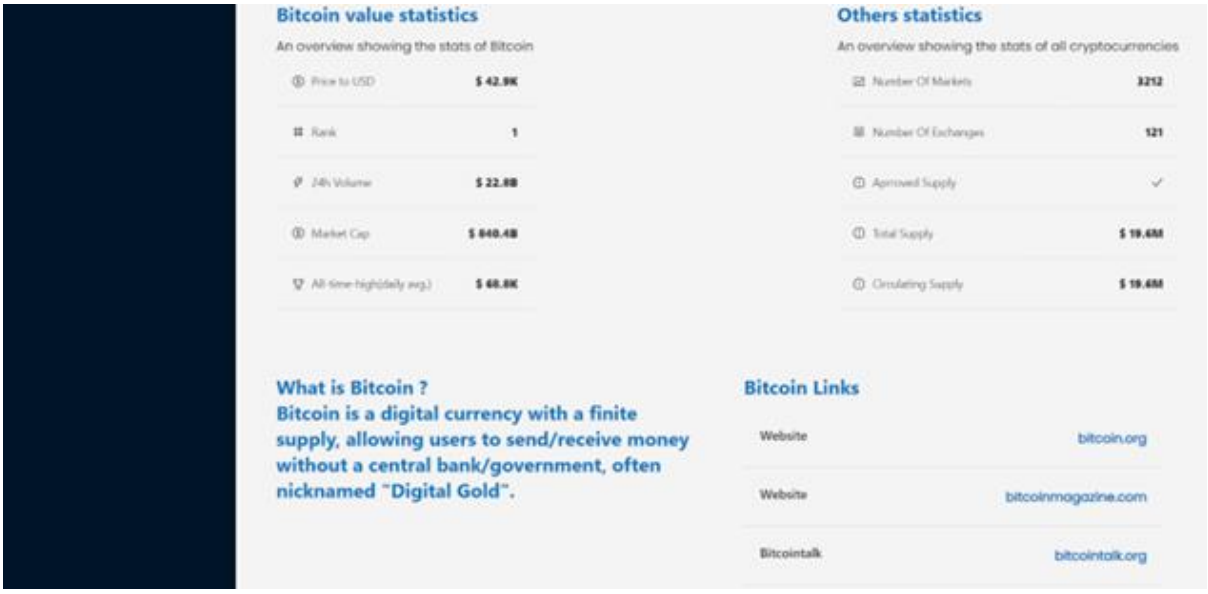
Crypto currency details page : This page contains the line chart with data representation of price of cryptocurrencies. Also contains statistics and website links of cryptocurrencies.



5y

Bitcoin Price Chart

Change: 1084.17%   Current Bitcoin Price: $ 42.9K



Bitcoin value statistics

An overview showing the stats of Bitcoin

Price to USD	$ 42.9K
Rank	1
24h Volume	$ 22.8B
Market Cap	$ 840.4B
All-time-high(daily avg.)	$ 68.8K

Others statistics

An overview showing the stats of all cryptocurrencies

Number Of Markets	3212
Number Of Exchanges	121
Approved Supply	✓
Total Supply	$ 19.6M
Circulating Supply	$ 19.6M

What is Bitcoin ?
Bitcoin is a digital currency with a finite supply, allowing users to send/receive money without a central bank/government, often nicknamed "Digital Gold".

Bitcoin Links

Website	bitcoin.org
Website	bitcoinmagazine.com
Bitcointalk	bitcointalk.org

Conclusion:

Each of these sections contributes to creating a cohesive and functional platform. The **Introduction** and **Description** provide clarity on what Cryptoverse offers. The **Scenario** and **Prerequisites** help users understand how they can engage with the platform. The **Project Structure** and **Project Flow** ensure that both the development and the user experience are seamless. Meanwhile, the **Cryptocurrency Component** and **Homepage** are essential features for making the platform both interactive and user-friendly.

This detailed breakdown ensures that Cryptoverse can cater to the needs of different user groups, from beginners to experts, while providing the tools to track and understand the cryptocurrency market.

Team Leader - Esha A - eshaakrishnan@gmail.com

Team Member 1 - Kiruthika G - kiruthikakiruthika273@gmail.com

Team member 2 - Aarthi G - aarthiguna2004@gmail.com

Team Member 3 - Guruvarshini J - guruvarshiniv@gmail.com